Mining spatial association rules in census data

Donato Malerba, Floriana Esposito, Francesca A. Lisi and Annalisa Appice

Dipartimento di Informatica, Università degli Studi di Bari,
Via Orabona, 4, I-70126 Bari
E-mail: malerba@di.uniba.it; esposito@di.uniba.i; lisi@di.uniba.it;
appice@di.uniba.it

Abstract

In this paper we propose a method for the discovery of spatial association rules, that is, association rules involving spatial relations among (spatial) objects. The method is based on a multi-relational data mining approach and takes advantage of the representation and reasoning techniques developed in the field of inductive logic programming (ILP). In particular, the expressive power of predicate logic is profitably used to represent spatial relations and background knowledge (such as spatial hierarchies and rules for spatial qualitative reasoning) in a very elegant, natural way. The integration of computational logics with efficient spatial database indexing and querying procedures permits applications that cannot be tackled by traditional statistical techniques in spatial data analysis. The proposed method has been implemented in the ILP system SPADA (spatial pattern discovery algorithm). We report the preliminary results of the application of SPADA to Stockport census data.

1. Background and motivation

Censuses make a huge variety of general statistical information on society available to both researchers and the general public. Population and economic census information is of great value in planning public services (education, funds allocation, public transportation), as well as in private businesses (locating new factories, shopping malls or banks, as well as marketing particular products).

The application of data mining techniques to census data, and more generally, to official data, has great potential in supporting good public policy and in underpinning the effective functioning of a democratic society [29]. Nevertheless, it is not straightforward and requires challenging methodological research, which is still in the initial stages.

As an illustrative example of some research issues, let us consider the census data table reported in Figure 1, where each row represents an enumeration district (ED), the smallest areal unit for which census data are published in UK ([1]).

([1]) National statistics institutes (NSIs) make a great effort to collect census data, but they are not the only organisations that analyse them: data analysis is often done by different institutes. By law, NSIs are prohibited from releasing individual responses to any other government agency or to any individual or business enterprise, so data are summarised for reasons of privacy before being distributed to external agencies and

c1	c24	c25	c26	c27	c28	c30	c32	c33	c34	c35	c36
03BSFA01	44	69	23	6	5	7	0	0	7	15	109
03BSFA02	56	108	36	8	11	22	0	2	12	27	233
03BSFA03	74	98	27	5	9	18	1	0	13	33	127
…	…	…	…	…	…	…	…	…	…	…	…

c1: ED level code, e.g. '03BSFA01', where '03' denotes a country/region (Greater Manchester), 'BS' denotes a district (Stockport), 'FA' denotes a ward (Bredbury) and '01' is the enumeration district.
c24: Total females of employees (full time) aged 16 and over
c25: Total males of employees (full time) aged 16 and over
c26: Total females of employees (part time) aged 16 and over
c27: Total males of employees (part time) aged 16 and over
c28: Total females of self-employed — with employees aged 16 and over
c30: Total males of self-employed — with employees aged 16 and over
c32: Total females of on a government scheme aged 16 and over
c33: Total males of on a government scheme aged 16 and over
c34: Total females of unemployed aged 16 and over
c35: Total males of unemployed aged 16 and over
c36: Total car availability in all households (households with three or more cars counted as having three cars)

Figure 1: An example of census data table. Data are summarised per enumeration district (ED)

The data analyst might be interested in finding some kind of dependence between the active population and the percentage of cars per household. A dependence can be expressed as an association rule, that is an implication of the form

$P \rightarrow Q$ (s %, c %),

where P and Q are a set of literals, called items, such that $P \cap Q = \emptyset$, while the percentages s % and c % are respectively called the support and the confidence of the rule, meaning that in s % of table rows both P and Q are true, and in c % of rows if P is true Q also holds. More formally, s estimates the probability $p(P \cup Q)$, while c estimates the probability $p(Q|P)$. The following is an example of an association rule establishing a dependence between the active population and the percentage of cars per household:

{low %FTEM, low %PTEM} → {low %PTEF, low %CH} (41 %, 62 %),

where low %FTEM, low %PTEM, low %PTEF, and low %CH are some items obtained by normalising and then discretising the attributes in Figure 1, namely:

low %FTEM: low (0.. 34 %) percentage of full-time employed males

institutes. Therefore, data analysts are confronted with the problem of processing data which summarise characteristics of groups of individuals.

low %PTEM: low (0.. 20 %) percentage of part-time employed males

low %PTEF: low (0.. 16 %) percentage of part-time employed females

low %CH: low (0.. 0.8 %) percentage of cars per household

For the sake of completeness, we report an alternative logical notation for the above association rule:

low %FTEM $\wedge$ low %PTEM $\rightarrow$ low %PTEF $\wedge$ low %CH (41 %, 62 %),

where the conjunction low %FTEM $\wedge$ low %PTEM $\wedge$ low %PTEF $\wedge$ low %CH is called pattern. This association rule states that in 62 % of EDs where there is both a low percentage of full-time employed males and a low percentage of part-time employed males, the percentage of part-time employed females is low and the percentage of cars per household is also low. The support is 41 %, meaning that in 41 % of analysed EDs all conditions expressed by the pattern low %FTEM $\wedge$ low %PTEM $\wedge$ low %PTEF $\wedge$ low %CH holds. By interpreting this association rule we can say that 41 % of EDs seem to be deprived areas.

1.1. The single table assumption

The discovery of association rules has attracted a great deal of attention in data mining research [11]. The blueprint for all the algorithms proposed in the literature is the levelwise method by Mannila and Toivonen [24], which is based on a breadth-first search in the lattice spanned by a generality order between patterns. Despite some interesting extensions, almost all algorithms reported in the literature share a restrictive data representation formalism, known as single-table assumption. More specifically, it is assumed that the data to be mined are represented in a single table (or relation) of a relational database, such that each row (or tuple) represents an independent unit of the sample population and the columns correspond to properties of units.

In some applications this assumption turns out to be a great limitation. For instance, in the above example, units correspond to EDs, which are spatial objects, since they have a geographical location. Having recognised this peculiarity, the data analyst may be interested in investigating the socioeconomic phenomenon of deprivation in association with the geographical distribution of EDs. To achieve this goal, the analyst may decide to augment the data table in Figure 1 with information on neighbouring units. In particular, for each ED in Figure 1, the analyst proposes the following data specifications:

- the number of schools in the neighbouring EDs,
- the number of banks in the neighbouring EDs, and
- the number of commercial activities in the neighbouring EDs,

since he/she suspects that the low percentage of cars can also be related to the number of services available in the neighbourhood.

If the analyst decides to represent the above data only for one neighbouring ED, the data table in Figure 1 can be extended by simply adding three attributes (see Table 1). What if he/she wants to represent the three attributes for all spatially adjacent EDs, which are variable in number? Under the single-table assumption he/she can create one entry for each adjacent ED in the original data table (see Table 2). However, this solution presents two main disadvantages:

(i) we have the usual problems connected with non-normalised tables, such as redundancy and anomalies in the insertion and removal of data.

(ii) we have one line per neighbouring ED, which means that the analysis results will really concern neighbouring EDs. In other words, the observation unit has deceptively changed.

Table 1: Three additional attributes of the nearest neighbour added to the single table

c1	**c24**	**c25**	**…**	**c36**	**Number of schools**	**Number of banks**	**Number of commercial activities**
03BSFA01	44	69	...	109	1	1	13
03BSFA02	56	108	…	233	0	0	23
03BSFA03	74	98	…	127	0	1	6
…	…	…	…	…	…	…	…

Table 2: Each row in the original table is duplicated to add information on a neighbouring ED

c1	**c24**	**c25**	**…**	**c36**	**Number of schools**	**Number of banks**	**Number of commercial activities**
03BSFA01	*44*	*69*	...	*109*	0	1	2
03BSFA01	*44*	*69*	...	*109*	1	0	3
03BSFA03	74	98	…	127	0	1	1
…	…	…	…	…	…	…	…

The former is a typical database issue, while the latter is more related to the data analysis procedure.

To solve these problems and keep the single-table assumption, the data analyst may try to summarise the information on the neighbouring EDs, say, by averaging the number of schools, banks and commercial activities (see Table 3). It is noteworthy that in this case there is no redundancy and standard data mining methods work well. However, there is an information loss that might lead to the misunderstanding of the underlying phenomenon. For instance, an ED can be adjacent to another ED with many services, as well as to other EDs

with no services at all, since they fall into the green belt of the city. By averaging the number of services per neighbouring ED, the analyst may give a totally wrong indication on the accessibility of services.

Table 3: Three additional attributes of the nearest neighbour added to the single table

c1	c24	c25	…	c36	Average number of schools	Average number of banks	Average number of commercial activities
03BSFA01	44	69	...	109	0.25	0.25	3.3
03BSFA02	56	108	…	233	0.33	0	0.36
03BSFA03	74	98	…	127	0	0.2	0.12
…	…	…	…	…	…	…	…

From a database perspective, the best representation of data would be that in Figure 2.

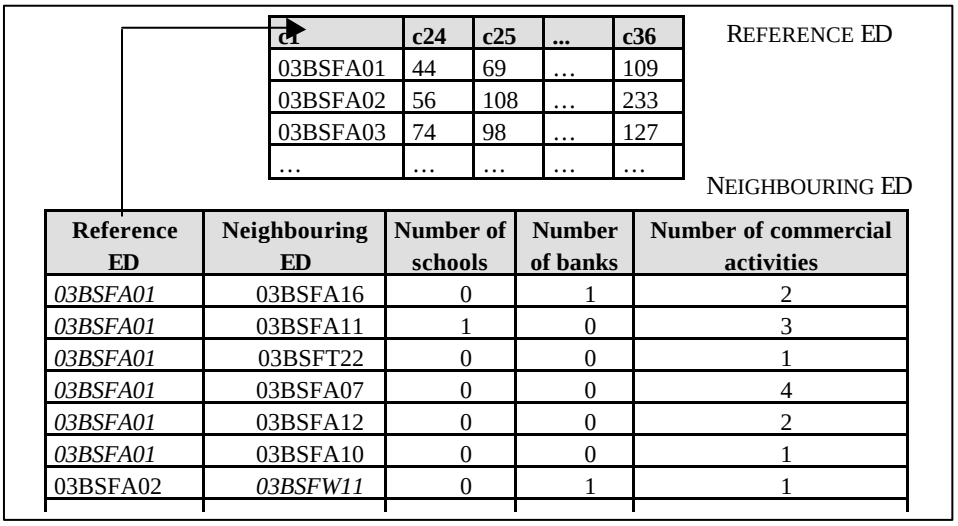

Figure 2: A multi-relation representation of socioeconomical attributes of some reference EDs and of their neighbouring ED. The attribute 'Reference ED' in the lower table is a foreign key of the upper table.

In this database two relations are defined, one for the reference EDs, that is, the EDs whose socioeconomical factors are the subject of investigation, and one for the neighbouring EDs, which are considered task relevant, because they are spatially adjacent to some reference EDs. Obviously, mining this simple database requires far more powerful methods which go beyond the single-table assumption.

1.2. A multi-relational data mining approach

The recently promoted **(multi-)relational** ([2]) approach to data mining [9] looks for patterns that involve multiple relations of a relational database. Thus the data taken as input by these approaches typically consists of several tables and not just a single one, as is the case in most existing data mining approaches. Patterns found by these approaches are called **relational** and are typically stated in a more expressive language than patterns defined in a single data table.

The following is an example of a **relational association rule**:

male-full-time-employee%(X,low) ∧ male-part-time-employee%(X,low) ∧
neighbour(X,Y) ∧ comm-activities(Y,high) → male-self-employee%(X,high)
(32 %, 70 %),

which states that in 70 % of the cases the low percentage of full-time and part-time male employees in some reference ED X, adjacent to another task relevant ED Y, with many commercial activities, implies a high percentage of self-employed males in X. The relational pattern

male-full-time-employee%(X,low) ∧ male-part-time-employee%(X,low) ∧
neighbour(X,Y) ∧ comm-activities(Y,high) ∧ male-self-employed%(X,high)

occurs in 32 % of reference EDs.

It is noteworthy that in this example, and more generally in relational association rules, the items are first-order logic **atoms**, that is, ***n*-ary predicates** applied to ***n* terms**. In this example terms can be either **variables**, such as X and Y, or **constants**, such as **low** or **high**. In other words, subsets of **first-order logic**, which is also called predicate calculus or relational logic, are used to express relational patterns and relational association rules.

This strong link with logics is not surprising, since any relational database can be easily modelled by a **deductive relational database** (DDB), by simply transforming all tuples in materialised tables into ground facts (extensional part of a DDB) and all views into rules (intensional part of a DDB) (see Figure 3).

([2]) The term multi-relational data mining is sometimes preferred to relational data mining and has also been used to denote data mining applied to relational databases [15].

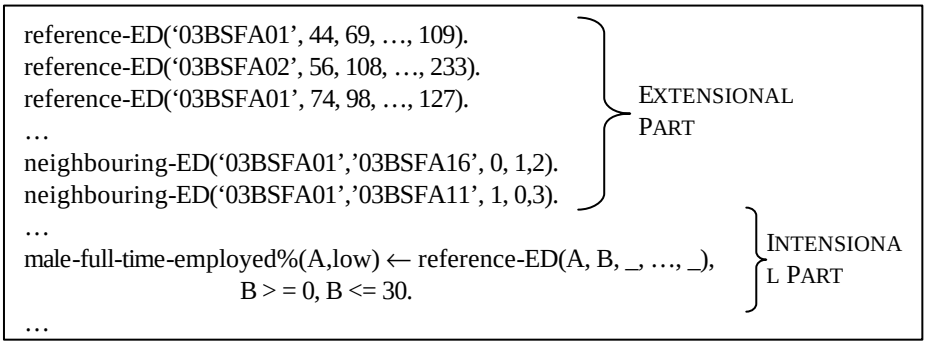

Figure 3: A deductive database view of the relational database in Figure 2. The extensional part is a set of ground facts corresponding to the tuples of the materialised tables, while the intensional part is a set of logical rules defining the views created in the relational database. In this example, we assume that the attribute ***male-full-time-employed%*** *has been defined by creating a view in the relational database.*

Therefore, a relational pattern is simply a DDB query, whose result set cardinality corresponds to the support.

Considering this strong link with logics, it is not surprising that many algorithms for multi-relational data mining originate from the field of **inductive logic programming** (ILP) [25], [8], [19], [26]. ILP has always been concerned with finding patterns expressed as logic programs. Initially, its main focus was on automated program synthesis from examples [2], but, in recent years, the scope of ILP has broadened to cover the whole spectrum of data mining tasks (association rules, regression, clustering and so on).

Extending a single-table data mining algorithm to a relational one is not trivial. Considerable insight and creativity is required to extend some key notions, such as distance measure and probabilistic dependence, to multi-relational data. Efficiency is also very important, as even testing a given relational pattern for validity is often computationally expensive. Moreover, for relational pattern languages, the number of possible patterns can be very large and it becomes necessary to limit their space of possible patterns by providing explicit constraints (**declarative bias**). These normally specify what relations should be involved in the patterns, how the relations may be interconnected and what other syntactic constraints the patterns have to obey.

1.3. Additional issues in spatial data mining

As explained above, there are two reasons for approaching the problem of mining spatial association rules as a multi-relational data mining problem. First, attributes of the neighbours of some spatial object of interest may influence the object itself, hence the need for representing object interactions. Second, different geographical objects may have different

properties, which can be properly modelled by as many data tables as the number of object types, hence the inadequacy of the single-table representation.

Some proposals for mining **relational** association rules have already been reported in literature [6]. However, mining **spatial** association rules is a more complex task. Two further degrees of complexity are:

(i) the implicit definition of spatial relations and

(ii) the granularity of the spatial objects.

The former is due to the fact that the location and the extension of spatial objects **implicitly** defines spatial relations such as topological, distance and direction relations. Therefore, complex data transformation processes are required to make spatial relations explicit (see the application of machine learning techniques to topographic map interpretation [22]).

The latter refers to the fact that spatial objects can be described at multiple levels of granularity. For instance, UK census data can be geo-referenced with respect to the following hierarchy:

$$\text{ED} \rightarrow \text{Ward} \rightarrow \text{District} \rightarrow \text{County},$$

based on the **inside** relationship between locations ([3]). Interesting rules are more likely to be discovered at low granularity levels (ED and ward) than at the county level. On the other hand, large support is more likely to exist at higher granularity levels (district and county) rather than at low levels.

In the next section, a new algorithm for mining spatial association rules is reported. The algorithm, named SPADA (**s**patial **pa**ttern **d**iscovery **a**lgorithm), is based on an ILP approach to relational data mining and permits the extraction of multi-level association rules, that is, association rules involving spatial objects at different granularity levels. SPADA has been implemented in Sictus Prolog and is interfaced to an Oracle8*i*® database, empowered by an Oracle Spatial cartridge, which enables spatial data to be stored, accessed and analysed quickly and efficiently. The system also performs the appropriate data transformation by extracting spatial features (Featex module) and by discretising numerical attributes (RUDE module). The application of SPADA to two data mining tasks involving UK census data is reported in Section 3.

([3]) In particular, the Stockport district of Greater Manchester is divided into 22 wards (Bredbury, Brinnington, Cale Green, Cheadle, Cheadle Hulme North, Cheadle Hulme South, Davenport, East Bramhall, Edgeley, Great Moor, Hazel Grove, Heald Green, Heaton Mersey, Heaton Moor, Manor, North Marple, North Reddish, Romiley, Shipping, South Marple, South Reddish, West Bramhall), each of which consists of 30 EDs on average.

2. Mining spatial association rules with SPADA

The discovery of spatial association rules is a descriptive mining task aiming to detect associations between **reference objects** and some **task-relevant objects**. The former are the main subject of the description, while the latter are spatial objects that are relevant for the task in hand and are spatially related to the former. For instance, we may be interested in describing a given area by finding associations between large towns (reference objects) and spatial objects belonging to the map layers of road network, hydrography and administrative boundaries (task-relevant objects). In particular, we look for **spatial** patterns, namely patterns that contain at least one spatial relationship. We call

$P \rightarrow Q$ (s %, c %)

a **spatial association rule**, if $P \cup Q$ is a spatial pattern.

As usual in the problem setting of association rule mining, we search for spatial associations with large support and high confidence (*strong rules*), such as

$$\text{is\_a}(X,\text{large\_town}) \wedge \text{intersects}(X,Y) \wedge \text{is\_a}(Y,\text{road}) \rightarrow$$
$$\text{intersects}(X,Z) \wedge \text{is\_a}(Z,\text{road}) \wedge Z \neq Y \qquad (91\ \%, 85\ \%),$$

which states that '**If** a large town X intersects a road Y, **then** X intersects a road Z distinct from Y **with** 91 % **support and** 85 % **confidence**'.

Since some kind of taxonomic knowledge of task-relevant geographic layers may also be taken into account to obtain descriptions at different granularity levels (**multiple-level association rules**), finer-grained answers to the above query are also expected, such as:

$$\text{is\_a}(X,\text{large\_town}) \wedge \text{intersects}(X,Y) \wedge \text{is\_a}(Y,\text{regional\_road}) \rightarrow$$
$$\text{intersects}(X,Z) \wedge \text{is\_a}(Z,\text{main\_trunk\_road}) \wedge Z \neq Y \qquad (45\ \%, 90\ \%),$$

which provides more insight into the nature of the task relevant objects Y and Z, according to the spatial hierarchy reported in Figure 4.

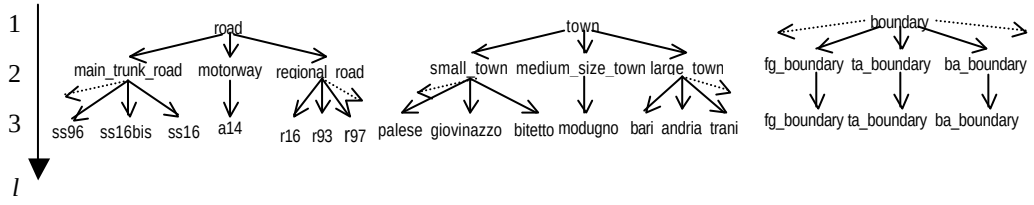

Figure 4: Three spatial hierarchies and their association to three granularity levels

It is noteworthy that the support and the confidence of the last rule changed. Generally, the lower the granularity level, the lower the support of association rules. Therefore, we follow Han and Fu's [14] proposal to use different thresholds of support and confidence for different granularity levels.
The problem of mining spatial association rules can be formally stated as follows:

Given:
- a spatial database (SDB),
- a set of reference objects S,
- some sets R_k, $1 \leq k \leq m$, of task-relevant objects
- some spatial hierarchies H_k involving objects in R_k
- M granularity levels in the descriptions (1 is the highest while M is the lowest)
- a set of granularity assignments ψ_k which associate each object in H_k with a granularity level
- a couple of thresholds *minsup*[l] and *minconf*[l] for each granularity level

Find: strong multi-level spatial association rules.

The problem has been already tackled by Koperski et al. [18]. They propose a top-down, progressive refinement method which exploits taxonomies both on spatial predicates (two-step spatial computation) and spatial objects (pattern discovery). The method has been implemented in the module Geo-associator of the spatial data mining system GeoMiner [16]. This method, however, suffers from severe limitations due to the single-table assumption. Our aim is to show the usefulness of an ILP approach to mining spatial association rules and, more generally, to spatial data mining. Representation problems and algorithmic issues related to the application of our logic-based computational method are discussed in the next two subsections.

2.1. The representation

The basic idea in our proposal is that a spatial database boils down to a deductive relational database (DDB) once the spatial relationships between reference objects and task-relevant objects have been extracted. The expressive power of first-order logic in databases also

allows us to specify background knowledge (BK), such as spatial hierarchies, **constraints** on spatial patterns and association rules (**declarative bias**), as well as **domain specific knowledge** expressed as sets of **rules**. In particular, the declarative bias helps to constrain the search in the exponentially large space of patterns, so that only interesting patterns are actually generated and evaluated. On the contrary, the specification of a domain specific knowledge allows SPADA to search for patterns which could not be otherwise found in the spatial database. The rules defining the domain specific knowledge are stored in the intensional part of the DDB and can support, amongst other things, spatial qualitative reasoning. The current version of SPADA supports the specification of both the declarative bias and the domain specific knowledge, which should be considered additional input to the system.

Henceforth, we denote the DDB in hand $D(S)$ to mean that it is obtained by adding the data extracted from SDB regarding the set of reference objects S to the previously supplied BK. The ground facts in $D(S)$ can be grouped into distinct subsets: each group, uniquely identified by the corresponding reference object $s \in S$, is called **spatial observation** and denoted $O[s]$. We define the set:

$$R[s] = \{r_i \mid \exists k: r_i \in R_k \text{ and a ground fact } \theta(s, r_i) \text{ exists in } D(S)\}$$

as the set of task-relevant objects related to s. The set $O[s]$ is given by

$$O[s] = O[s|s] \cup \bigcup_{r_i \in R[s]} O[r_i \mid s]$$

where:

- $O[s|s]$ contains properties of s and spatial relations between s and r_i
- $O[r_i|s]$ contains properties of r_i and spatial relations between r_i and some $s' \in S$.

In an extreme case, $O[s]$ can coincide with $D(S)$. This is the case in which s is spatially related to all task-relevant objects. The unique reference object associated to a spatial observation allows us to define the support and the confidence of a spatial association rule. More precisely, the spatial association rule $P \rightarrow Q$ (s %, c %) means that in s % of spatial observations both conjunctions P and Q hold and in c % of spatial observations where P is true Q holds too. Note that the notion of spatial observation in SPADA adapts the notion of **interpretation**, which is common to many relational data mining systems [9], to the case of spatial databases.

Example 1: Suppose the mining task is to the discover the associations relating large towns (S) with waterways (R_1), roads (R_2) and province boundaries (R_3) in the Province of Bari, Italy. We are also given a BK including the spatial hierarchies of interest and three levels of granularity (see Figure 4).

hierarchy(town, 1, null, [town]).

hierarchy(town, 2, town, [large_town, medium_size_town, small_town]).
hierarchy(town, 3, large_town, [bari, altamura, andria, barletta, trani, bitonto, molfetta, gravina, monopoli, corato, gioia_del_colle]).
hierarchy(town, 3, medium_size_town, [modugno, palo_del_colle, terlizzi, ruvo, noicattaro, adelfia, grumo, giovinazzo, mola_di_bari]).
hierarchy(town, 3, small_town, [palese, bitetto, binetto, toritto, valenzano, cassano, mariotto, palombaio]).
hierarchy(road, 1, null, [road]).
hierarchy(road, 2, road, [motorway, main_trunk_road, regional_road]).
hierarchy(road, 3, motorway, [a14]).
hierarchy(road, 3, main_trunk_road, [ss16, ss16bis, ss96, ss98, ss99, ss100]).
hierarchy(road, 3, regional_road, [r16, r93, r97, r170, r171, r172, r271, r378]).
hierarchy(water, 1, null, [water]).
hierarchy(water, 2, water, [sea, river]).
hierarchy(water, 3, sea, [adriatico]).
hierarchy(water, 3, river, [ofanto, lacone]).
hierarchy(boundary, 1, null, [boundary]).
hierarchy(boundary, 2, boundary, [fg_boundary, ta_boundary, br_boundary, mt_boundary, pz_boundary]).
is_a(*X*, *Y*):- hierarchy(_, _, *Y*, Nodes), member(*X*, Nodes).
is_a(*X*, *Y*):- hierarchy(Root, _, Father, Nodes), member(*X*, Nodes), is_a(Father, *Y*).

Here, the **is_a** stands for an **instance_of** relation between spatial objects and their geographical layers. Spatial relations between objects in S and objects in any of R_1, R_2 and R_3, are extracted by means of spatial computation and transformed into facts of the kind <spatial relation>(RefObj, TaskRelevantObj) to be added to $D(S)$.

Spatial observations are portions of $D(S)$, each concerning a reference object. In our case, there are 11 distinct spatial observations, one for each large town. For instance, O[barletta] is given by the union of the sets of ground facts listed in Table 4. By definition, the observation encompasses not only spatial relationships between the reference object barletta$\in S$ and task-relevant objects in R_1 (adriatico etc.), R_2 (a14 etc.), R_3 (fg_boundary etc.), but also spatial relationships between each of these task-relevant objects and some other $s' \in S$ (e.g. giovinazzo) like in adjacent_to(giovinazzo, adriatico).

Let $A = \{a_1, a_2, \ldots, a_t\}$ be a set of atoms whose terms are either variables or constants (Datalog atoms [4]). Predicate symbols used for A are all those permitted by the user-specified declarative bias, while the constants are only those defined in $D(S)$. The atom denoting the reference objects is called **key atom**. For instance, with reference to the above example of the Province of Bari, A contains the key atom is_a(*X*,large_town), 'spatial' atoms such as close_to(*X*,*Y*), intersects(*X*,*Y*), and adjacent_to(*X*,*Y*), and 'taxonomic' atoms such as is_a(*X*,road), is_a(*X*,main_trunk_road), …, is_a(*X*,water), is_a(*X*,sea).

Conjunctions of atoms on A are called **atomsets** [5] like the item sets in classical association rules. In our framework, a language of patterns $L[l]$ at the granularity level l is a set of well-formed atomsets generated on A. Necessary conditions for an atom set P to be in $L[l]$ are the presence of the key atom, the presence of 'taxonomic' atoms exclusively at the granularity level l, the linkedness [17] and the safety. In particular, the last property guarantees the correct evaluation of patterns when the handling of negation is required (see Example 2). To a pattern P we assign an existentially quantified conjunctive formula $eqc(P)$ obtained by turning P into a Datalog query.

Table 4: The spatial observation O[barletta]

***O*[barletta | barletta]**	***O*[a14 | barletta]**	***O*[r170 | barletta]**
is_a(barletta, large_town).	is_a(a14, road).	is_a(r170, road).
adjacent_to(barletta, adriatico).	intersects(bari, a14).	intersects(andria, r170).
Intersects(barletta, a14).	intersects(trani, a14).	**...**
Intersects(barletta, ss16).	intersects(bitonto, a14).	***O*[r193 | barletta]**
Intersects(barletta, ss16bis).	intersects(gioia_del_colle, a14).	is_a(r193, road).
Intersects(barletta, r170).	intersects(molfetta, a14).	**...**
Intersects(barletta, r193).	**...**	*O*[fg_boundary | barletta]
close_to(barletta, fg_boundary).	***O*[ss16 | barletta]**	is_a(fg_boundary, boundary).
...	is_a(ss16, road).	adjacent_to(trani, fg_boundary).
***O*[adriatico | barletta]**	intersects(bari, ss16).	…
is_a(adriatico, water).	intersects(trani, ss16).	*O*[ss16bis | barletta]
adjacent_to(bari, adriatico).	intersects(monopoli, ss16).	is_a(ss16bis, road).
adjacent_to(trani, adriatico).	intersects(molfetta, ss16).	intersects(bari, ss16bis).
adjacent_to(molfetta, adriatico).	…	intersects(trani, ss16bis).
adjacent_to(giovinazzo, adriatico).		intersects(molfetta, ss16bis).
…		…

Definition: A pattern P covers an observation $O[s]$ if $eqc(P)$ is true in $O[s] \cup BK$.

Example 2: The pattern

$$P \equiv \text{is\_a}(X, \text{large\_town}), \text{intersects}(X,R), \text{intersects}(Y,R), Y \backslash= X, \text{is\_a}(R, \text{road})$$

covers the spatial observation O[barletta] shown in Table 1 because the corresponding

$$\text{eqc}(P) \equiv \exists\ \text{is\_a}(X, \text{large\_town}) \wedge \text{intersects}(X,R) \wedge \text{intersects}(Y,R) \wedge Y \backslash = X \wedge \text{is\_a}(R, \text{road})$$

is satisfied by O[barletta]$\cup$BK. Here the predicate \= is the ISO prolog standard built-in predicate for the non-unifiability of two variables. Note that it hides a negation.

Definition: Let O be the set of spatial observations in $D(S)$ and O_P denote the subset of O containing the spatial observations covered by the pattern P. The support of P is defined as $\sigma(P) = | O_P | / |O|$.

Definition: A spatial association rule in $D(S)$ at the granularity level l is an implication of the form

$P \rightarrow Q$ (s %, c %),

where $P \cup Q \in L[l]$, $P \cap Q = \emptyset$, P includes the key atom and at least one spatial relationship is in $P \cup Q$. The percentages s and c are respectively called the support and the confidence of the rule, meaning that s % of spatial observations in $D(S)$ is covered by $P \cup Q$ and c % of spatial observations in $D(S)$ that is covered by P is also covered by $P \cup Q$.

Definition: The support and the confidence of a spatial association rule $P \rightarrow Q$ are given by

$s = \sigma(P \cup Q)$ and $c = \varphi(Q|P) = \sigma(P \cup R) / \sigma(P)$.

In multi-level association rule mining, an *ancestor* relation between two patterns at different granularity levels $P \in L[l]$ and $P' \in L[l']$, $l < l'$ exists if and only if P' can be obtained from P by replacing each spatial object $h \in H_k$ at granularity level $l = \psi_k(h)$ with a spatial object $h' < h$ in H_k, which is associated with the granularity level $l' = \psi_k(h')$.

The frequency of a pattern depends on the granularity level of task-relevant spatial objects.

Definition: Let $minsup[l]$ and $minconf[l]$ be two thresholds setting the minimum support and the minimum confidence respectively at granularity level l. A pattern P is **large** (or frequent) at level l if $\sigma(P) \geq minsup[l]$ and all ancestors of P with respect to the hierarchies H_k are large at their corresponding levels. The confidence of a spatial association rule $P \rightarrow Q$ is high at level l if $\varphi(Q|P) \geq minconf[l]$. A spatial association rule $P \rightarrow Q$ is **strong** at level l if $P \cup Q$ is large and the confidence is high at level l.

2.2 Method

The task of mining spatial association rules itself can be split into two sub-subtasks:

(i) find large (or frequent) spatial patterns;

(ii) generate highly-confident spatial association rules.

The reason for such a division is that frequent patterns are not commonly considered useful for presentation to the user as such. They can be efficiently post-processed into rules that exceed given threshold values. In the case of association rules the threshold values of support and confidence offer a natural way of pruning weak, rare rules.

Algorithm design for frequent pattern discovery has turned out to be a popular topic in data mining. Most algorithms proposed in the literature are based on a breadth-first search in the lattice spanned by a generality order ≥ between patterns. Given two patterns P_1 and P_2, we write $P_1 \geq P_2$ to denote that P_1 is more general than P_2 or equivalently that P_2 is more specific than P_1. The space is searched one level at a time, starting from the most general patterns and iterating between the candidate generation and candidate evaluation phases. The high-level algorithm of SPADA implements the aforementioned levelwise method (see Figure 5).

Cycle on the level (l ³ 1) of the spatial hierarchies
  Find large 1-atomsets at level l
  Cycle on the depth (k > 1) of search in the pattern space
    Generate candidate k-atomsets at level l from large (k-1)-atomsets
    Generate large k-atomsets at level l from candidate k-atomsets
  Until the user-defined maximum depth
Until the user-defined maximum granularity level M

Figure 5: A high-level view of the levelwise mining algorithm SPADA.

The pattern space is structured according to the θ-subsumption [28]. Many ILP systems adopt θ-subsumption as the generality order for clause spaces. In this context we need to adapt the framework to the case of atomsets. More precisely, the restriction of θ-subsumption to **Datalog queries** (i.e. existentially quantified conjunctions of Datalog atoms) is of particular interest.

Definition: Let Q_1 and Q_2 be two queries. Then Q_1 q-**subsumes** Q_2 if and only if there exists a substitution θ such that $Q_1 \supseteq Q_2\theta$.

Example 3: Let us consider the queries

$Q_1 \equiv \exists$ is_a(X, large_town) $\wedge$ intersects(X, R) $\wedge$ is_a(R, road)
$Q_2 \equiv \exists$ is_a(X, large_town) $\wedge$ intersects(X,Y)
$Q_3 \equiv \exists$ is_a(X, large_town)

We say that Q_1 θ-subsumes Q_2 and Q_2 θ-subsumes Q_3 with substitutions $\theta_1=\{Y\backslash R\}$ and $\theta_2=\varnothing$ respectively.

We can now introduce the generality order adopted in SPADA.

Definition: Let P_1 and P_2 be two patterns. Then P_1 is more general than P_2 under θ-subsumption, denoted as $P_1 \geq_\theta P_2$, if and only if P_2 θ-subsumes P_1.

A graphical representation of the lattice spanned by $\geq_\theta$ including the queries reported in example 3 is shown in Figure 6.

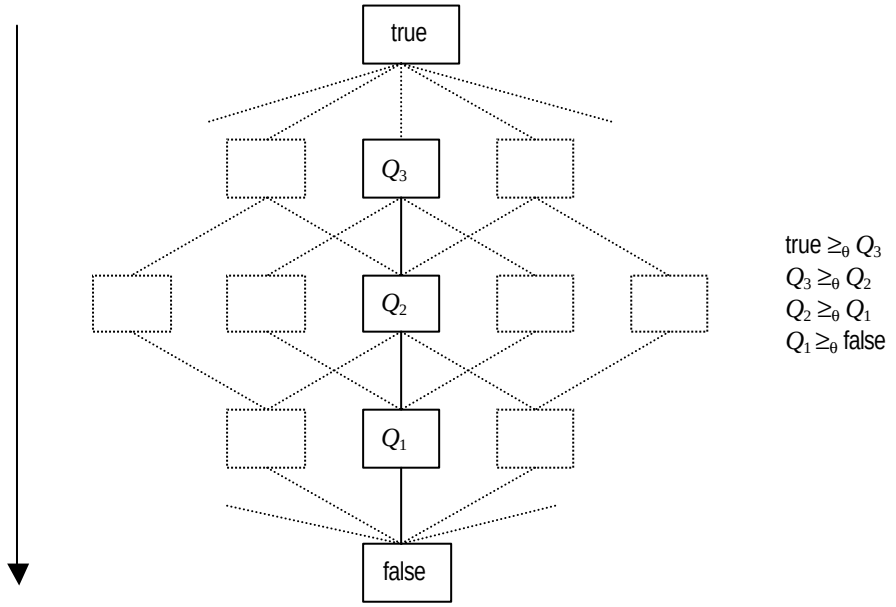

Figure 6: Example of pattern space structured according to $^{3}_{q}$

For θ-subsumption the following properties hold:

- reflexivity: $P \geq_\theta P$;
- transitivity: $P_1 \geq_\theta P_2$ and $P_2 \geq_\theta P_3$, then $P_1 \geq_\theta P_3$;
- decidability: a procedure exists to decide if $P_1 \geq_\theta P_2$.

The anti-symmetric property does not hold for θ-subsumption, therefore θ-subsumption is a **quasi-ordering**. It follows that, given two queries such that $P_1 \geq_\theta P_2$ and $P_2 \geq_\theta P_1$, we cannot conclude that P_1 and P_2 are equal up to renaming, i.e. P_1 and P_2 are not alphabetic variants ([4]). As shown below, this feature has to be taken into account during the search.

A quasi-ordered set of patterns can be searched by a **refinement operator**, namely a function which computes a set of refinements of a pattern. In particular, we need a refinement operator under θ-subsumption that enables the bottom-up search of the pattern space from the most specific to the most general patterns.

Definition: Let $<G, \geq_\theta>$ be a pattern space ordered according to $\geq_\theta$. A **downward refinement operator under** q**-subsumption** is a function ρ such that $\rho(P) \subseteq \{Q \mid P \geq_\theta Q\}$.

([4]) Let E and F be two expressions. Then E and F are **variants**, denoted $E \approx F$, if and only if substitutions θ and σ exist such that $E = F\theta$ and $F = E\sigma$. We also say that E is an **alphabetic variant** of F. For instance, $f(X)$ and $f(Y)$ are alphabetic variants.

It is noteworthy that $\geq_\theta$ on patterns represented as Datalog queries is monotone with respect to support, which is the criterion for candidate evaluation in SPADA. Therefore, the refinement operator drives the search towards patterns with decreasing support. Moreover, all refinements $\rho(P)$ of an infrequent pattern P are infrequent. This is the first-order counterpart of one of the properties holding in the family of the a priori-like algorithms [1], on which the pruning criterion is based.

For each granularity level (l), SPADA generates and evaluates candidates by searching the pattern space. The **candidate generation** phase consists of a refinement step followed by a pruning step. The former applies the refinement operator under θ-subsumption to patterns previously found to be frequent by preserving the property of linkedness [17]. The latter mainly involves verifying that candidate patterns do not θ-subsume any infrequent pattern. Further pruning criteria have been implemented in SPADA. In particular, the system checks that candidates are not alphabetic variants of previously discovered patterns. The complexity of this test is $O(n^2)$, where n is the number of atoms in the two patterns to be compared. However, this test is performed an exponential number of times, thus making the overall computational cost very high. Solutions have been proposed by Nijssen and Kok [27] to gain better performances in the general case of relational association rules. In the context of multiple-level relational association rules, different strategies have been identified by Lisi and Malerba [20]. The **candidate evaluation** phase is performed by comparing the support of the candidate pattern with the minimum support threshold set for the level being explored. If the pattern turns out not to be a large one, it is rejected. As for the support count, the candidate is transformed into an existential query whose answer set supplies all the substitutions that make the pattern true in $D(S)$. In particular, the number of different bindings for the variable which is the placeholder for reference objects is assumed as the absolute frequency of the pattern in $D(S)$.

A rough preliminary remark on the computational complexity of SPADA leads to the notorious trade-off between expressiveness and efficiency in first-order representations. Indeed, it is well known that a simple matching of two expressions with commutative and associative operators (such as the logical OR of atoms in a clause) is NP-complete [12]. Therefore, any known algorithm that checks the coverage of an atom set or that equivalently evaluates a query with respect to a relational database has an exponential complexity. Nevertheless, it has also been proved that queries with up to k atoms, where each atom contains at most j terms, can be evaluated in polynomial time [7]. Whether these constraints are applicable to the domain of spatial data analysis is still under investigation.

Related to efficiency is **scalability**. Indeed, studies on the learnability theory have shown that current ILP algorithms would scale relatively well as the number of examples or facts in the background knowledge increases. However, they would not scale well with the number of arguments of the predicates (relations) involved, and in some cases with the complexity of the patterns being searched. The use of **declarative bias** is usually suggested to improve scalability. It is a set of constraints on spatial patterns and association

rules that guide the application of the refinement operator ρ during the candidate generation phase. Indeed, a refinement step consists of adding one or more atoms from $L[l]$ to the pattern to be refined. The more restrictions we put on the patterns, the smaller the search space, and hence the faster its search. In general, there is a trade-off between the efficiency of an ILP system and the quality of the patterns it comes up with.

2.3. Integrating SPADA with other software components

The application of the ILP approach to spatial databases is made possible by a middle-layer module for feature extraction, as shown in Figure 7.

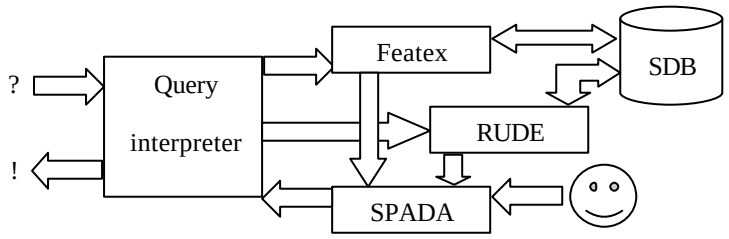

Figure 7: Integration of SPADA with other software modules which support spatial feature extraction (Featex) and discretisation of numerical features (RUDE). Additional input to SPADA, such as declarative bias and background knowledge, is directly provided by the user.

This layer is essential to cope with one of the main issues of spatial data mining, namely the requirement of complex data transformation processes to make spatial relations explicit.

This function is partially supported by the spatial database (SDB), which offers spatial data types in its data model and query language and supports them in its implementation, providing at least spatial indexing and efficient algorithms for spatial join [13]. Thus spatial databases supply an adequate representation of both single objects and spatially related collections of objects. In particular, the abstraction primitives for spatial objects are point, line and region. Among the operations defined on spatial objects, spatial relationships are the most important because they make it possible, for example, to ask for all objects in a given relationship with a query object. Egenhofer and Herring [10] proposed the nine-intersection model to categorise binary topological relations between arbitrary spatial objects. Examples are the relation **meet** between two regions and the relation **crosses** between a region and a line. The nine-intersection model is implemented in the Oracle Spatial cartridge to support the computation of some topological relations.

Many spatial features (relations and attributes) can be extracted from spatial objects stored in SDB. They can be categorised as follows:

(i) geometric, that is, based on the principles of Euclidean geometry;

(ii) directional, that is, regarding relative spatial orientation in two or three dimensions;

(iii) topological, that is, binary relations that preserve themselves under topological transformations such as translation, rotation and scaling;

(iv) hybrid, that is, features which merge properties of two or more of the previous three categories.

This variety requires for the development of a feature extractor module, named Featex, which also enables the coupling of SPADA with the SDB. Featex is implemented as an Oracle package of procedures and functions implemented in the PL-SQL language. In this way, it is possible to formulate complex SQL queries involving both spatial and aspatial data (e.g. census data). The set of spatial features that can be extracted by this module is reported in Table 5.

Since SPADA, like many other association rule mining algorithms, cannot process numerical data properly, it is necessary to perform a discretisation of numerical features with a relatively large domain. For this purpose, we have implemented the relative unsupervised discretisation algorithm RUDE [21] which proves to be suitable for dealing with numerical data in the context of association rule mining. At the end of all this data processing, query results are stored in temporary database tables. An ad hoc PL-SQL function transforms these tuples into ground Datalog facts of $D(S)$.

Table 5: Spatial features extracted by the feature extractor module

Feature	Meaning	Type	Values
almost_parallel(*Y*,*Z*)	Parallelism relation between *Y* and *Z*	Hybrid relation	{true, false}
almost_perpendicular (*Y*,*Z*)	Perpendicularity relation between *Y* and *Z*	Hybrid relation	{true, false}
density(*Y*,*Z*)	Area(*Y*)/Area(*Z*)	Hybrid relation	Real
direction(*Y*)	Geographic direction of object *Y*	Directional attribute	{north, east, north_west, north_east}
distance(*Y*,*Z*)	Distance between *Y* and *Z*	Geometrical relation	Real
layer_name(*Y*)	Object *Y* type	Aspatial attribute	Layer name
line_shape(*Y*)	Object *Y* shape	Geometrical attribute	{Straight, curvilinear}
relate(*Y*,*Z*)	Topological relation between *Y* and *Z*	Topological attribute	Type of topological relation

3. Application to Stockport census data

In the context of the SPIN! project we investigated the application of spatial data mining techniques to some issues reported in the unitary development plans (UDP) of Stockport, one of the 10 metropolitan districts of Greater Manchester, United Kingdom.

3.1. The data

Spatial analysis is made possible by the use of the Ordnance Survey's digital maps of the district, where several interesting layers are available, namely ED/ward/district boundaries, roads, bus priority lanes, and so on. In particular, Stockport is divided into 22 wards for a total of 589 EDs. By joining UK 1991 census data available at the ED summarisation level with ED spatial objects, it is possible to investigate socioeconomic issues from a spatial viewpoint. In total 89 tables, each having 120 attributes on average, have been made available for policy analysis. Census attributes provide statistics on the population (resident and present at the census time, ethnic group, age, marital status, economic position, and so on), on the households in each ED (number of households with *n* children, number of households with *n* economically inactive people, number of households with two cars, and so on) as well as on some services available in each ED (e.g. number of schools).

For the application of our spatial association rule mining method we have focused our attention on transportation planning, which is one of the key issues in UDP. In the following subsection, we report results for the problem of characterising the area crossed by the M63 motorway. For another application to the accessibility of the area around the Stepping Hill Hospital, see the paper by Malerba et al. [23].

3.2. Characterising the area crossed by the M63 motorway

One of the problems is a decision-making process concerning the M63 motorway. More precisely, we are asked to describe the area of Stockport served by the M63 (i.e. the wards of Brinnington, Cheadle, Edgeley, Heaton Mersey, South Reddish) from the sociological viewpoint, in order to provide some hints for transport planners. The data considered in this analysis concerns census statistics on commuters. The description of the area is expressed by some spatial association rules at two levels of granularity. A hierarchy for the Stockport ED layer has been obtained by grouping EDs on the basis of the ward they belong to (see

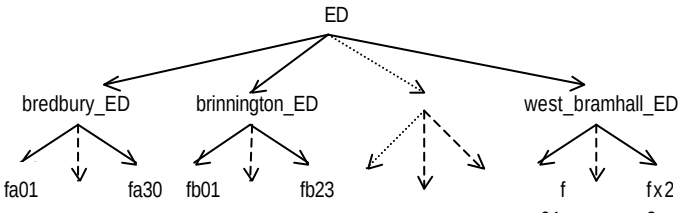

Figure 8: An is_a hierarchy for the Stockport ED layer

Figure 8) and expressed as Datalog facts in BK.

Spatial association rules should relate EDs crossed by the M63 (reference objects) to EDs in the area served by the M63 (task-relevant objects) (see Figure 9).

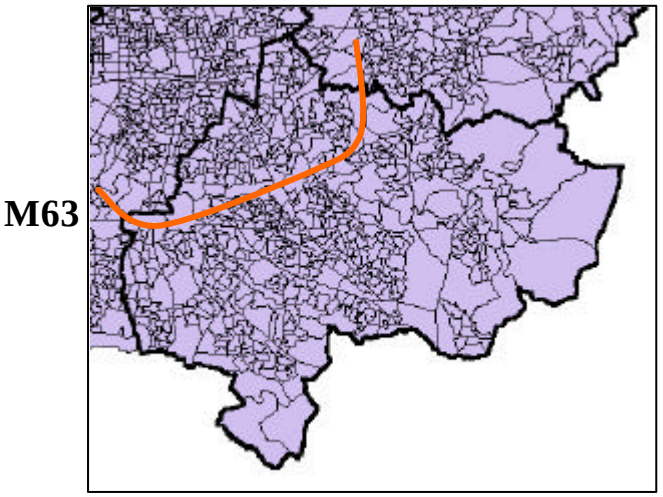

Figure 9: Stockport district and its EDs crossed by the M63 motorway

The relations of intersection (EDs–motorways) and adjacency (EDs–EDs) have been extracted for the area of interest and transformed into Datalog facts of $D(S)$. The following census attributes have been selected for this experiment:

- s820161, persons who work outside the district of residence and drive to work;
- s820213, employees and self-employed workers who reside in households with three or more cars and drive to work;
- s820221, employees and self-employed workers who reside in households with three or more cars and work outside the district of residence.

Since they refer to residents aged 16 and over, they have been normalised with respect to the total number of residents aged 16 and over (s820001). Moreover, they have been discretised by RUDE, since they are all numeric (more precisely, integer valued). At the end of this transformation process, each ED is described by three ground atoms in $D(S)$, namely dr_out(X, [a..b]), cars3_dr(X, [a..b]), cars3_out(X, [a..b]), where X denotes an ED, while [a..b] is one of the intervals returned by RUDE.

The key atom defining the reference objects in S is ed_on_M63(X), which is intensionally defined in the BK by means of the following rule:

ed_on_M63(X):— intersect(X, m63)

The BK also includes the declarative specification of some rules for spatial qualitative reasoning, namely

can_reach(X, Y):— intersect(X, m63), intersect(Y, m63), Y\=X.

close_to(X, Y):— adjacent_to(X, Z), adjacent_to(Z, Y), Y\=X.

Finally, the following thresholds for support and confidence were defined: min_sup[1] = 0.7 and min_conf[1] = 0.9 at the first level, and min_sup[2] = 0.5 and min_conf[2] = 0.8 at the second level.

SPADA was run on the $D(S)$ obtained. The runtime was 331 seconds for association rules at granularity level 1, and 310 seconds for level 2 (data refers to a Pentium III 1 GHz PC with 256 Mb RAM).

Initially, the system returned 12 925 frequent patterns out of 74 338 candidate patterns, for a total of 12 466 strong rules. By analysing them we observed that some were actually useless, since they did not relate spatial data to census data. In other words, some association rules were pure spatial patterns, such as the following:

ed_on_M63(X), can_reach(X,Y) → is_a(Y,ward_on_m63_ED) (90.0 %, 100.0 %),

which states that if an ED (Y) in the area served by the M63 can be reached from an ED crossed by the M63, then that ED is certainly (100 % confidence) an ED of a ward crossed by the M63. Despite the high support and confidence, this pure spatial pattern is of no interest for transport planners.

In a second run, we decided to constrain the search to patterns containing at least one of the census attributes dr_out(*X*, [a..b]), cars3_dr(*X*, [a..b]) and cars3_out(*X*, [a..b]). This is possible by specifying the following declarative bias:

pattern_constraint([dr_out(_,_),cars3_dr(_,_),cars3_out(_,_)],1)

where the first argument of the predicate *pattern_constraint* is the list of atoms to include in the relational pattern, while the second argument is the minimum number of required atoms of the list.

The system generated 10 513 strong association rules in 1520 seconds (time increased because of constraint checking for each generated pattern). Some of them have a very high support and confidence and provide the expert with some hints on the habits of commuters, such as the following association rule discovered at level 2:

ed_on_M63(*X*), close_to(*X*,*Y*), is_a(*Y*,Bedgeley_ED) →
cars3_out(*X*,[0.0..0.037]), cars3_dr(*X*,[0.0..0.037]) (100 %, 100 %),

which states that 'if an ED crossed by the M63 (*X*) is close to another ED of the ward of Bedgeley (*Y*), then in that ED the percentage of people living in households with three or more cars and driving out of the district to work is very low (less than 4 %)'. It is important to point out that this is simply an association and does not define any kind of cause–effect relationship between the place where people live and their social habits. Another interesting spatial association rule at the same granularity level is the following:

ed_on_M63(*X*), can_reach(*X*,*Y*) → is_a(*Y*,heaton_mersey_ED),
dr_out(*Y*,[0.2857..0.4782]), cars3_out(*Y*,[0.0..0,037]) (80.0 %, 88.88 %),

which states that 'if an ED *Y* in the M63 area can be reached from another one crossed by the M63 motorway (*X*), then it is in the Heaton Mersey ward and has quite a high percentage of people that drive to work but do not live in households with three or more cars'.

Finally, we decided to constrain the search space further, by asking only for those spatial patterns involving EDs where people have the same commuting habits. This time the first argument of the predicate **pattern_constraint** is a list of sub-lists, where each sub-list denotes a conjunction of atoms to be included in the relational patterns. In particular, we have defined the following declarative bias:

pattern_constraint([[dr_out(*X*,*Z*), dr_out(*Y*,*Z*), *X*\=*Y*],
[cars3_dr(*X*,*Z*), cars3_dr(*Y*,*Z*),*X*\=*Y*], [cars3_out(*X*,*Z*), cars3_out(*Y*,*Z*),*X*\=Y]], 1).

SPADA found only 345 strong rules (79 for level 1 and 266 for level 2) in about 833 seconds. The following is an example of association found by the system at the granularity level:

ed_on_M63(A) → can_reach(A,B), is_a(B,cheadle_ED), can_reach(A,C),$C \neq B$, is_a(C,edgeley_ED), cars3_dr(C,[0.0..0.037]), cars3_dr(B,[0.0..0.037])
(90 %, 90 %),

which states that from an ED crossed by the M63 it is possible to reach (by the same motorway) two EDs, one in Cheadle and one in Edgley, with the same low percentage of people living in families with three or more cars and driving out of the district to work.

4. Conclusions

In the above application, we have seen that some of the discovered rules actually convey new knowledge. However, the search for these 'nuggets' requires a lot of tuning and efforts by the data analyst in order to constrain the search space properly and discard most of the obvious or totally useless patterns hidden in the data. This is typical of exploratory data analysis and SPADA can be considered one of the most advanced tools that data analysts currently use in their iterative knowledge discovery process.

One of the main limitations of SPADA, which is also a problem of many other relational data mining algorithms, is the requirement of some expertise in data and knowledge engineering. Indeed, the user should know how data are organised in the spatial database (e.g. layers and physical representation of objects), the semantics of spatial relations that can be extracted from digital maps, the meaning of some parameters used in the discretisation process and in the generation of spatial association rules, as well as the correct and most efficient way to specify the domain knowledge and declarative bias. In future work, we will investigate some 'interestingness measures' of rules for presentation purposes, so that the user can browse the output XML file of spatial association rules as simply as possible. In addition, we intend to study the relation with 'symbolic data analysis' [3] and the possibility of using the software developed in the context of the SODAS project for the analysis, summarisation and visualisation of rules obtained by generalising spatial objects covered by some spatial association rules returned by SPADA.

5. Acknowledgements

The authors thank Jim Petch, Keith Cole and Mohammed Islam (MIMAS, University of Manchester, England) and Chrissie Gibson (Department of Environmental and Geographical Sciences, Manchester Metropolitan University, England) for providing access to census data and digital OS maps of Stockport Manchester. The work presented in this paper is in partial fulfilment of the research objectives set by the IST European project SPIN! (**Sp**atial mining for data of public **in**terest) and by the MURST COFIN-2001 project

'Methods for the extraction, validation and representation of statistical information in a decision context'. Thanks to Lynn Rudd for her help in reading the paper.

6. References

[1] Agrawal, R. and Srikant, R., 'Fast algorithms for mining association rules', *Proceedings of the 20th VLDB conference*, Santiago, Chile, 1994.

[2] Bergadano, F. and Gunetti, D., *Inductive logic programming: from machine learning to software engineering*, The MIT Press, Cambridge, MA, 1996.

[3] Bock, H. H. and Diday, E. (eds.), *Analysis of symbolic data — Exploratory methods for extracting statistical information from complex data*, Studies in classification, data analysis, and knowledge organisation series, Vol. 15, Springer-Verlag, Berlin, 2000.

[4] Ceri, S., Gottlob, G. and Tanca, L., 'What you always wanted to know about Datalog (and never dared to ask)', *IEEE transactions on knowledge and data engineering*, Vol. 1, No 1, 1989, pp. 146–166.

[5] Dehaspe, L. and De Raedt, L., 'Mining association rules in multiple relations', Lavrac, N and Dzeroski, S. (eds), *Inductive logic programming*, LNCS 1297, Springer-Verlag, Berlin, 1997, pp. 125–132.

[6] Dehaspe, L. and Toivonen, H., 'Discovery of frequent Datalog patterns', *Data mining and knowledge discovery*, Vol. 3, No 1, 1999, pp. 7–36.

[7] De Raedt L. and Dzeroski, S., 'First order jk-clausal theories are PAC-learnable' *Artificial Intelligence*, Vol. 70, 1994, pp. 375–392.

[8] De Raedt, L., *Interactive theory revision*, Academic Press, London, 1992.

[9] Dzeroski, S. and Lavrac, N. (eds), *Relational data mining*, Springer-Verlag, Berlin, 2001.

[10] Egenhofer, M. J. and Herring, J. R., 'Categorising binary topological relations between regions, lines, and points in geographic databases', Egenhofer, M. J., Mark, D. M. and Herring, J. R. (eds.), *The nine intersection: formalism and its use for natural-language spatial predicates*, 1994, pp. 183–271.

[11] Fayyad, U. M., Piatetsky-Shapiro, G. and Smyth, P., 'From data mining to knowledge discovery: an overview', Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P. and Uthurusamy, R. (eds), *Advances in knowledge discovery in databases*, AAAI Press/The MIT Press, 1996, pp. 1–34.

[12] Garey, M. R. and Johnson, D. S., *Computers and intractability*, W. H. Freeman and Co., San Francisco, California, 1979.

[13] Güting, R. H., 'An introduction to spatial database systems', *VLDB Journal*, Vol. 3, No 4, 1994, pp. 357–399.

[14] Han, J. and Fu, Y., 'Discovery of multiple-level association rules from large databases', Dayal, U., Gray, P. M. D. and Nishio, S. (eds), *VLDB'95 — Proceedings of the 21st international conference on very large databases*, Morgan-Kaufmann, 1995, pp. 420–431.

[15] Han, J., Fu, Y., Wang, W., Chiang, J., Gong, W., Koperski, K., Li, D., Lu, Y., Rajan, A., Stefanovic, N., Xia, B. and Zajane, O. R., 'DBMiner: a system for mining knowledge in large relational databases', *Proceedings of the 1996 international conference on data mining and knowledge discovery (KDD'96)*, Portland, Oregon, 1996, pp. 250–255.

[16] Han, J., Koperski, K., Stefanovic, N., 'GeoMiner: a system prototype for spatial data mining', Peckham, J. (ed.), *Sigmod 1997 — Proceedings of the ACM–Sigmod international conference on management of data*, Sigmod, Record 26, No 2, 1997, pp. 553–556.

[17] Helft, N., 'Inductive generalisation: a logical framework', Bratko, I. and Lavrac, N. (eds), *Progress in machine learning*, Sigma Press, 1987, pp. 149–157.

[18] Koperski, K., Adhikary, J. and Han, J., 'Spatial data mining: progress and challenges', *Proceedings of the workshop on research issues on data mining and knowledge discovery*, Montreal, Canada, 1996.

[19] Lavrac, N. and Dzeroski, S., *Inductive logic programming: techniques and applications*, Ellis Horwood, Chichester, 1994.

[20] Lisi, F. and Malerba, D., 'Efficient discovery of multiple-level patterns', *Atti del Decimo Convegno Nazionale su Sistemi Evoluti per Basi di Dati SEBD 2002*, 2002, pp. 237–250.

[21] Ludl, M.-C. and Widmer, G., 'Relative unsupervised discretisation for association rule mining', Zighed, D. A., Komorowski, H. J. and Zytkow, J. M. (eds), *Principles of data mining and knowledge discovery*, LNCS 1910, Springer-Verlag, 2000, pp. 148–158.

[22] Malerba, D., Esposito, F., Lanza, A. and Lisi, F. A., 'Machine learning for information extraction from topographic maps', Miller, H. J. and Han, J. (eds), *Geographic data mining and knowledge discovery*, Taylor and Francis, London, 2001, pp. 291–314.

[23] Malerba, D., Lisi, F. A., Appice, A. and Sblendorio, F., 'Mining spatial association rules in census data: a relational approach', *Proceedings of the ECML/PKDD'02 workshop on mining official data*, University Printing House, Helsinki, 2002, pp. 80–93.

[24] Mannila, H. and Toivonen, H., Levelwise search and borders of theories in knowledge discovery, *Data mining and knowledge discovery*, Vol. 1, No 3, 1997, pp. 259–289.

[25] Muggleton, S. (ed), *Inductive logic programming*, Academic Press, London, 1992.

[26] Nienhuys-Cheng, S.-H. and deWolf, R., *Foundations of inductive logic programming*, Springer, Heidelberg, Germany, 1997.

[27] Nijssen, S. and Kok, J. N., 'Faster association rules for multiple relations', Nebel, B. (ed), *Proceedings of the 17th international joint conference on artificial intelligence*, Morgan Kaufmann, 2001, pp. 891–896.

[28] Plotkin, G., 'A note on inductive generalisation', *Machine intelligence*, No 5, 1970, pp. 153–163.

[29] Saporta, G., 'Data mining and official statistics', *Atti della Quinta Conferenza Nazionale di Statistica*, Rome, 2000, pp. 15–17.